

\$2.00



Contents:

Editorial	2
Index to advertisers	3
April Competition	4
News from Barsons	4
Wordwise hints and problems	6
Music Program mod.	6
Help	7
Integrity and reality	8
Function Keys	12
Hangman	13
Random Selection	18
Proc star	19
Inverse Video ptII	23
Instant Arcade Games - review	26
Looney Lift - review	26
Beebug Subscription	27
Advanced Programming-review	28
Beyond Basic - review	29
S.G.M.	30
Hall of fame	31
Local Meetings	32

THE BBC MICROCOMPUTER USERS GROUP OF NZ INC

NEWSLETTER

VOLUME 2 NUMBER 2 MARCH 1984

Not only is our previous Editor a wonderful person with a formidable array of wit, charm and imagination - she is also better looking than me. If it were not for the tremendous amount of work Anne has put into this newsletter (including this issue) and its associated systems, and the fact that I have the power and flexibility of the Beeb to call on, there would have been no way anyone could have talked me into this job! The development of the newsletter to its current level, and the parallel development of probably the best organised and most dynamic computer club in the country, is in no small way attributable to Anne's work. She got the odd bit of assistance from others such as GCF of course....

Anne somehow also managed to run a household - which frequently seemed to include a good proportion of the local Beeb community. The youngest members of the family seem to have had a flying introduction to computer literacy, albeit perhaps at the expense of other aspects of their education. Young John was down here the other week and, observing that my Beeb was connected into the colour TV set, announced that he didn't "like that program". He then proceeded to try and banish the cricket test with the aid of the Keyboard....

HELP !

Anne maintained high standards throughout the Beeblet. I intend to continue those standards. To do this I will need the support of ALL of you. It is very important that the newsletter continues to receive a wide range of material in terms of both levels of sophistication and areas of interest.

Recently a friend picked up a copy of the Beeblet, flipped through it, and then murmured, "Can't understand a word" Do any of you feel like that about the newsletter ? I do not see the Beeblet as the vehicle for a small circle of Beeb experts. Rather it should be a forum for the whole group - including the complete beginner.

This means YOU letting me know what you are interested in, what you are doing, what you would like to hear about and what problems you are having. There is a lot happening around the country which is of interest to the wider membership - interesting people passing through, a wide range of seminars and workshops organised in the various regions and the Beeb moving into schools. At the moment I only get a trickle of often second hand news about many such events.

A lot of people can learn from your experiences and discoveries. It does not have to be a full blown article if a few short sentences in a brief letter would do. The monthly letter promised by Joe Joyce of Barsons is a major step in the right direction - and who knows what a bit of feedback might achieve.

PROGRAMMING STYLE

A lot of fuss has been made in recent months about "structured programming" - the current buzzword amongst the people-who-know. Actually, there is a lot of sense in it. Especially when you consider that a major function of this newsletter is to communicate ideas and techniques. The technique provides a simple but systematic approach, which is self documenting. The result, whether from a novice or a high-priest, should be an ordered flow of logic, simple to follow and maintain. I will therefore arrange a series of articles to provide guidelines for all you programmers - both current and aspiring. Did I here some heckling from the cheap seats? All contributions, views or even abuse will be gratefully received.

SUBSCRIPTION LEVEL

I have had a tiny trickle of letters coming in about the proposed sub increase. Take it that most of you are happy? There will be one more issue (mag), before the AGM, for you to air your views.

PAPER

Any comments on the new glossy look? No, the papers not any more expensive than that used pre February. Resists coffee stains and is easier for the printers though!

PROMOTION

The longer the print run, the lower the unit cost of Beeblet - a major component of your subscription. If you do not like the mag, tell me. If you do, tell your friends. Does your local dealer/computer bookshop sell Beeblets? If not, why not?

ADVERTISERS INDEX:

Business World	20
FG Software	17
G.H. Bennett	27
Molymerx	25
Pirimai Pharmacy	31

NEXT MONTH:

More on disk drives
Structured Programming
Games?
Book Reviews
Software Reviews
Hardware Reviews?

APRIL COMPETITION

The National Committee, in its wisdom, has left me a fairly free hand this month as regards a suitable competition. There may be one prize or several (up to a total value of \$50), depending on the number and quality of entries. You may thus stand to win something more than just the usual box of cassette tapes!

The topics this month are:

1. Exciting, useful or just plain fiendish function key definitions.
2. April 1st thoughts, articles or programs (these should be labelled appropriately to warn me what is going on.) It has been suggested that some Acorn news releases would make a good model....

Entries will be assessed on the basis of usefulness, wit and elegance.

NEWS FROM BARSONS

"Probably the best news of the year for the BBC, though not for existing owners, is the announcement of a 15% decrease for a standard cassette based machine."

Joe Joyce, Acorn Products Manager with Barson Computers, has agreed to write a short column for each edition of the magazine giving news of new software or peripherals and answer queries, criticisms or comments about Barsons or Acorn.

In the February note, which unfortunately missed our deadline, Joe suggested that many of the problems which in 1983 beset distributors, retailers and the Beeb orientated public, have been resolved. He also noted the visits of Herman Hauser and Owen Maddock (Sales Manager for Acorn International) : promises of a much improved Acorn performance re product development and realistic delivery dates. Joe also

mentioned that he had been invited to speak at an Auckland U.G. meeting (.. Auckland - of interest to others?) and that Beeb networks were beginning to move into N.Z. schools - even in Wellington.

HARDWARE

In his March letter Joe notes :

"I have been asked to comment on a report in 'Acorn Update' regarding supply of 1.2 operating systems. While Barson Australia have been able to upgrade to 1.2 at no cost, we are unable to do the same. Our charge of \$28 covers the cost of freight from the U.K. and the duty and customs clearance fees we are charged on arrival in N.Z.

We have limited quantities of Basic 2 ROMs in stock and can offer them to members of the Users group for \$40." First come first served, further supplies expected in five to six weeks.

Barsons also have supplies of the P.I. digitiser (advertised in Acorn User) and the LVL double density D.O.S. (NB. The review of the LVL double density DOS in Acorn User Feb 84 is very harsh. 80 track drives, standard in N.Z. Beeb systems, do not appear to have adequate software support. The reviewer also noted compatibility problems with other software. A review, or comments, from someone who has installed the LVL D.D. D.O.S. would be welcome. - Ed.)

MAGAZINES

The price increases of 'Acorn User' and 'Micro User', distributed by Barsons, are a direct result of higher airfreight charges. These reflect the rapidly increasing size of these magazines. Surface mail would be cheaper, but involves a 2 month delay.

Joe notes that he has been trying to get copies of 'A&B Computing' for six months, but hopes to have supplies by April.

EDUCATION

A.E.U.G. (N.Z.) is a new magazine launched by a group of Auckland teachers. Joe says " the Acorn Education User's Group was established in Melbourne at the end of 1982 and has built up a large and dedicated following among Australian schoolteachers. Working closely with Melbourne, a group of teachers in Auckland have organised the N.Z. branch and invite new members nationally. They intend to publish a public domain software catalogue for BBC education users and to make the programs available on disc for a nominal sum. Enquiries should be directed to either Peter Revell, at Rangitoto College, Auckland, or to Ms. Jackie Tyrrell at Auckland Girls Grammar."

WORDWISE HINTS AND PROBLEMS

David J. Dunedin

David has mentioned in passing that:

-It helps to configure the computer not to send linefeeds to the printer (ie. rely on the printer auto-linefeed if it has this) using *fx6,10. Use this also to send the codes for the printer to cancel skip-over-perforation (again assuming this feature is already set by internal switches in the printer). Both these allow correct conditional paging and begin new page functions since WORDWISE appears not to use formfeed instructions but rather a series of blank lines until the number of lines set per page is reached, before the formfeed is activated.

-The lack of a visible <cr> symbol in WORDWISE is frustrating. Does anyone know how to change this?

-Another useful revelation has been the use of *BUILD to make up a mailing list data file, then using *EXEC within a program to use the file for printing out labels for instance.

-Does anyone have any secrets for using this sort of thing in similar fashion to GF"xxx" for inserting addresses into standard letters with WORDWISE ?

MUSIC PROGRAM MOD

Neil has pointed out that in restructuring his music program (published in the February issue) we have inadvertently eliminated a useful feature - namely the facility to repeat the same tune without having to enter the speed each time. He suggests:

```
line 105 cpm=100
      1175 PRINT "Speed=";cpm
      1150 (delete)
      1340 PRINT "6....etc" "9 change speed" "10 save or end"
      1455 IF A=9 PROCsetspeed:Procmusic
      1460 UNTIL A>9
```

He also suggests:

line 110 Instructions are not required if you are playing only. Would be better renumbered 185

870 Does strange things on his cassette based system - "RECORD THEN RETURN" is printed automatically, see the UG p193 line 11005

1180 delete (duplicates line 1155)

HELP WANTED

A CRY FOR HELP
from Mark, Wellington.

Do you live in the Kelburn area and have a Beeb with a disc drive? I may be forced to sell mine soon for financial reasons and I am in the middle of several projects. If anyone could give me access to their Beeb on the odd evening for a couple of hours that would be greatly appreciated. Please ring me at home, 757 980.

TORCHPACK OWNERS

Glyn of Waikanae would like to hear from other local BBC(B)-280 Torchpack users. Phone PRAM 36391

GALACTIC INVADERS

A.D.W. of Napier has the "Galactic Invaders" program from the September 83 Micro User which works perfectly so long as he leaves out the cursor delete command. If he includes it then upon firing a rocket a mid-air explosion is produced, followed by the message "Subscript at line 600". The magazine did not provide instructions or explanations and error tracing exercises have proved fruitless.

HELP GIVEN

NOUGHTS AND CROSSES PROBLEM

ADW also suggests that line 1070 (VDU 23;8202;0;0;0;) of the noughts and crosses program might cause strange effects. Anton suggests:

Be very careful to put in semicolons in VDU commands, not commas. The reason is that a semicolon means "send this as a two byte number". So "0;" is sent as "0,0". If you only have a comma, the computer will keep taking bytes until it has the required number. The current line may work, but the subsequent line could well be corrupted.

Checking your listing backwards (forcing a character by character examination) is a painful but effective way of picking up such problems. It can save weeks of error tracing - Ed.

INTEGRALITY AND REALITY

Anton

BASIC quite properly distinguishes between reals (ie. "real" numbers, which include fractions) and integers. So, on occasion, do mothers - say when counting children.

While a women can say she is slightly pregnant, that is really a joke. Pregnancy is integral. Statisticians, on the other hand, can deal in reals and say that the average nubile women is 7.2% pregnant (Just a guess, folks!).

Time is another real. Given two moments, no matter how close, a third moment can fit in between.

Reals do not work on computers, which are very categorical machines indeed. So the designers cheat and say : this categorical number (call it A) is nearly the number you want. "Nearly" varies from computer to computer. The Beeb is pretty accurate (one part in a billion). Interestingly, the least significant byte (of 5 used) is rounded to the nearest correct value rather than truncated. The following program, which prints out the five bytes (in hex) of the real variable A, shows that the least significant byte (right hand column) is rounded up for 8 and above.

10 MODE7	80 0 0 0 0
20 2%=&203	80 1 11 11 11
30 FOR I%=120 TO 135	80 2 22 22 22
40 A=I%/240	80 3 33 33 33
50 FOR J%=3 TO 7	80 4 44 44 44
60 PRINT ~J%?LOMEM;	80 5 55 55 55
70 NEXT	80 6 66 66 66
80 PRINT	80 7 77 77 77
90 NEXT	80 8 88 88 89
	80 9 99 99 9A
	80 A AA AA AB
	80 B BB BB BC
	80 C CC CC CD
	80 D DD DD DE
	80 E EE EE EF
	80 10 0 0 0

Columns represent increasing values of the variable A.
 If column 1 = a, column 2 = b,....column 5 = e
 then $A = 2*(0.5 + \frac{b}{256} + \frac{c}{(256)^2} + \frac{d}{(256)^3} + \frac{e}{(256)^4})$.

IMPROVING YOUR INPUT

Dave C.

The INPUT command in BASIC is a very useful instruction but to my mind a little bit crude in its operation. The main limitations are that it has no way of restricting either the number or the type of characters entered, and gives no indication to the user as to what is expected. This can result in a nicely formatted input screen looking like a disaster area after Junior has leaned on the keyboard and you omitted to turn off auto repeat!

The following assembler routine makes use of the OSWORD (A=0) routine which contrary to the User Guide can be used by all operating systems including O.S. 0.1. The OSWORD routine operates in the same manner as INPUT but allows the input data to be restricted to a specified length and nominated range of ASCII characters. This on the face of it would appear to handle most of the requirements. The problem is that while OSWORD only inserts valid characters into the input buffer, it still continues to print anything that is input, valid or invalid! Fortunately OSWORD in common with all operating system routines uses the standard OSWRCH routine for printing characters on the screen. I have therefore inserted a routine to intercept calls to OSWRCH made by OSWORD in order to filter out the offending characters. Any invalid characters are replaced by the BELL control code (ASCII 7) which has the effect of not only suppressing the printing of the character but also sounds the alarm as the key is pressed.

The other main function of this subroutine is to print an input field of the appropriate length to indicate to the user the number of characters required. This is done by defining a pseudo blank character for ASCII code 255 in modes 0 to 6. The same code in mode 7 prints a very similar block type character.

The subroutine is called from BASIC as follows:

```
A%=length: X%=value1: Y%=value2: CALL Input
```

where length is the length of the input field (max 256) and value1 and value2 are the minimum and maximum acceptable ASCII values respectively. The routine always returns a string terminated by a carriage return. This is accessed by \$Buffer. In the case of numeric fields, VAL(\$Buffer) will convert the string into the appropriate internal format. The operation of Input is identical to INPUT. The DELETE, COPY, and cursor control keys carry out their normal editing functions except that DELETE inserts a pseudo blank instead of a space. CTRL U also carries out its normal function of deleting the whole field except that again pseudo blanks are inserted instead of

spaces. RETURN terminates the routine as for INPUT.

The following describes how Input works:

- 160-200 Define pseudo blank character, set up 5 byte parameter definition block, allocate memory for machine code and define fixed addresses.
- 240-290 This is the main Input routine which is called from BASIC as described above.
- 250 Stores user supplied parameters into parameter block.
- 260 Prints field of required length and repositions cursor.
- 270 Replaces contents of WRCHV with address of user routine.
- 280 Sets-up address of parameter block and calls OSWORD.
- 290 Restores contents of WRCHV to its normal value.
- 300-310 This is a subroutine used to print the input field and reposition the cursor to the start of the field.
- 320-370 This is the routine which intercepts OSWRCH call issued by OSWORD.
- 320 Needed for CTRL U function.
- 330 All control codes except delete passed directly to OS.
- 340 Prints pseudo blanks for DELETE and CTRL U functions.
- 350 Checks character is within valid ASCII range.
- 360 Invalid characters are replaced by BELL.
- 370 Jumps to O/S write character routine.

The remaining lines (10-120) are not part of the routine but are included to demonstrate its use. The machine code is first assembled and then a 3 digit numeric field is displayed for input. This is followed by a 15 character alphanumeric field limited to capitals only, i.e. lower case not allowed.

Notice that in both cases the input string returned is immediately assigned to another variable. This is because Input uses the same buffer address as INPUT at address &700. This is one of BASIC'S general work areas, the contents of which cannot be guaranteed once further instructions are executed.

The effects of line 30 help to illustrate the operation of the routine and will become apparent on running the program! An expanded assembler listing can be obtained by changing STEP in line 210 to 3.

```

10 MODE 6
20 PROCassemble
30 ?&D8=&30: ?&25A=&30
40 PRINT TAB(0,10);"Please enter your age ";
50 A/=3:X%=ASC"0":Y%=ASC"9":CALL Input
60 age%=VAL($Buffer)
70 PRINT TAB(0,12);"and place of birth ";
80 PRINT TAB(0,13);"in (block capitals). ";
90 A/=15:X%=ASC" ":Y%=ASC"Z":CALL Input
100 town$=$Buffer
110 PRINT/"I see you are ";age%;" years old"/"and were born in ";town$;".
120 END
130
140
150 DEF PROCassemble
160 VDU 23,255,0,0,0,126,126,126
170 DIM CSECT 101, PDB 5
180 WRCHV=&20E: Buffer=&700: OSWORD=&FFF1
190 WRCHRTN=?WRCHV+?(WRCHV+1)*256
200 ?PDB=Buffer MOD 256: ?(PDB+1)=Buffer DIV 256
210 FOR I%=0 TO 3 STEP 2
220   P%=CSECT
230   IOPT I%
240   .Input
250   STA PDB+2:STX PDB+3:STY PDB+4
260   LDA #255:JSR print:LDA #8:JSR print
270   LDA #wrchar MOD 256:STA WRCHV:LDA #wrchar DIV 256:STA WRCHV+1
280   LDA #0:LDX #PDB MOD 256:LDY #PDB DIV 256:JSR OSWORD
290   LDA exit+1:STA WRCHV:LDA exit+2:STA WRCHV+1:RTS
300   .print:LDX PDB+2
310   .loop:JSR WRCHRTN:DEX:BNE loop:RTS
320   .wrchar:CMP #8:BNE ctiltest:LDA #&7F
330   .ctiltest:CMP #&20:BCC exit:CMP #&7F:BNE rangetest
340   JSR WRCHRTN:LDA #255:JSR WRCHRTN:LDA #8:BNE exit
350   .rangetest:CMP PDB+3:BCC error:CMP PDB+4:BCC exit:BEQ exit
360   .error:LDA #7
370   .exit:JMP WRCHRTN:}
380 NEXT
390 ENDPROC

```

MAKING those FUNCTION KEYS work for you

Neville T. Ch'ch

REQUIRES OS 1.2

It is fairly well known by now that the teletext control codes 128-137 can be generated using SHIFT in conjunction with the red function keys. For example, holding down SHIFT + f1 will generate code 129, so all following text on that line will appear in RED.

These codes continue to be generated even if the keys have been previously defined using the *KEY commands. However, it is possible to generate ALL the teletext control codes except two (why two can't, will be explained later), through the use of SHIFT and CTRL + function keys, and also the cursor keys with the addition of *FX4,2

The codes can be generated as follows:

CHR#	KEYS	EFFECT
128	SHIFT + f0	Nothing
129	SHIFT + f1	alpha red
130	SHIFT + f2	alpha green
131	SHIFT + f3	alpha yellow
132	SHIFT + f4	alpha blue
133	SHIFT + f5	alpha magenta
134	SHIFT + f6	alpha cyan
135	SHIFT + f7	alpha white
136	SHIFT + f8	flash on
137	SHIFT + f9	flash off
139	*FX4,2+SHIFT+COPY	Nothing
140	*FX4,2+SHIFT++	Normal height
141	*FX4,2+SHIFT++	double height
142	*FX4,2+SHIFT+~	Nothing
143	*FX4,2+SHIFT++	Nothing
144	CTRL + f0	Nothing
145	CTRL + f1	graphic red
146	CTRL + f2	graphic green
147	CTRL + f3	graphic yellow
148	CTRL + f4	graphic blue
149	CTRL + f5	graphic magenta
150	CTRL + f6	graphic cyan
151	CTRL + f7	graphic white
152	CTRL + f8	concealed graphics
153	CTRL + f9	continuous graphics
155	*FX4,2+CTRL+COPY	Nothing
156	*FX4,2+CTRL++	switch to black background
157	*FX4,2+CTRL++	select new background
158	*FX4,2+CTRL+~	enable hold character
159	*FX4,2+CTRL++	release hold character

NOTES:

1. To generate codes 139-143, and 155-159, you must first type *FX4,2 to disable the cursor keys, and use the 4 arrow or COPY keys in conjunction with SHIFT or CTRL.
2. Where effect is 'Nothing', the control code has no effect.
3. Effects of all control codes in Mode 7 are summarised p.489 User Guide.

So, from this table, you can see that all the teletext control codes can be generated except CHR\$128 and CHR\$155. This is because f10 (which can be redefined), is the BREAK key. As you may have noticed, the control codes follow the keys in numeric order - eg SHIFT+f1=129, SHIFT+f2=130 etc - and so logically with f9=137, f10(BREAK) should give CHR\$138. However, although you can press SHIFT+BREAK, you get a hard reset of the Beeb!! Thus codes 138 (SHIFT+BREAK) and 154 (CTRL+BREAK) are inaccessible to us.

You can get interesting effects in Mode 7 by using lower case (CAPS LOCK off), pressing CTRL+f(x) to get a graphic colour, and typing in lower case letters - the Beeb prints the graphics characters out!. For a summary of these graphics characters, refer U.G.p.489.

HANGMAN

Ron M. Marlborough

Here is a version of the word guessing game "Hangman". You can run it as it stands but as the game is popular, parents will have to learn to change the vocabulary to prevent cunning progeny learning the answers off by heart. Ron's program allows such changes to be made easily. A straight word for word swap in the DATA is simple, and children's names, streets and towns could well go in. To add quite a lot of extra words, increase the word array in line 80 to say (200).

The program uses the flags END1 and END2 to mark the easy and difficult word groups. So to insert some easy words, use an extra line. 1755DATA BELINDA,..... Note the use of capitals. For simplicity, lower case is avoided.

That's all there is to it. Good, structured programs invite change, and make it easy to achieve.

Remember to put the semicolons in the VDU statements. The program auto starts on any error, such as ESCAPE, and you may have disabled BREAK as well. Great for young children, bad for LIST or for SAVE. Change line 200 to ON ERROR STOP until you are happy the program runs. Then reset line 200, and SAVE before you RUN it. LOAD again to make changes.

HAVE ---! (an easy word, in line 1750)

```

10 *****
20 REM -----HANGMAN-----
30 REM -----By Ron Manning-----
40 REM -----
50 REM -----Version 2--Nov.83-----
60 *****
70 MODE 7
80 DIM word$(160),music(8)
90 A$=""
100 FOR note=1 TO 8
110   READ music(note)
120 NEXT
130 N%=1
140 REPEAT
150   READ word$(N%)
160   IF word$(N%)="END1" THEN N%=(N%-1):easy%=(N%
170     N%=(N%+1
180   UNTIL word$(N%+1)="END2"
190   hard%=(N%-2
200 ON ERROR MODE 7
210 FOR N=1 TO 2
220   PRINT TAB(13,2+N);:PRINT CHR$(141);CHR$(130);"HANGMAN"
230   NEXT N
240 PRINT TAB(1,6);CHR$(131);"I will choose a word and you have to"
250 PRINT CHR$(131);"try and guess what it is.The number of"
260 PRINT CHR$(131);"letters in the word are indicated by"
270 PRINT CHR$(131);"the lines on the screen.Each time you"
280 PRINT CHR$(131);"get a letter wrong I will make a part"
290 PRINT CHR$(131);"of your gallows. You are allowed twelve"
300 PRINT CHR$(131);"wrong attempts, before you get hung."
310 PRINT"" Press the SPACE BAR to continue"
320 VDU 23;8202;0;0;0;
330 REPEAT UNTIL GET=32
340 CLS
350 PRINT TAB(1,9);CHR$(131);"ENTER ";CHR$(135);"1 ";CHR$(131);"(for young chil
dren's words)"
360 PRINT TAB(3,11);CHR$(131);"OR ";CHR$(135);"2 ";CHR$(131);"(for older chil
dren's words)"
370 PRINT TAB(13,12);CHR$(131);"(and for mums and dads)"
380 PRINT TAB(1,20);"Press -ESCAPE- at any time to change"
390 PRINT TAB(6,21);"the degree of difficulty"
400 REPEAT
410   menu$=GET$
420   A%=VAL(menu$)
430   UNTIL A%=1 OR A%=2
440 MODE 5
450 *****

```

```

460 REPEAT
470   CLS
480   VDU 19,0,0,0,0,0
490   VDU 23;8202;0;0;0;
500   VDU 24,600;8;1279;500;
510   COLOUR 129:CLS
520   GCOL 0,130:CLG
530   GCOL 0,0
540   IF A%=1 THEN wstart%=0:wend%=
easy%
550   IF A%=2 THEN wstart%=easy%:we
nd%=hard%
560   REM ** Choose a random word *
*
570   word=RND(wend%-wstart%)+wstar
t%
580   REM ** Print # of letter line
s **
590   IF A$(word$(word)) THEN A$=wo
rd$(word)
600   PRINT TAB(4,8);
610   FOR LY=1 TO LEN(A$)
620     PRINT "-";
630     NEXT LY
640   REM ** Compare letters **
650   PRINT TAB(2,3)*"Choose a lette
r"
660   try=1
670   game=0:good=0:hang=0
680   U$=""
690   hang=0
700   REPEAT
710     REPEAT
720       X$=GET$
730       UNTIL X$="A" AND X$<="Z"
740       *FX15,0
750       guess=0
760       IF INSTR(A$,X$) AND INSTR(U
$,X$)=0 THEN PROCright
770       IF guess=0 THEN PROCwrong
780       IF good=<(LY-1) THEN PROCgo
od
790       UNTIL game=1
800     UNTIL FALSE
810   END
820
830 DEF PROCgallows
840   IF hang=0 THEN MOVE 770,50:DRAW
1030,50
850   IF hang=1 THEN MOVE 900,50:DRAW
900,380
860   IF hang=2 THEN MOVE 800,50:DRAW
900,150
870   IF hang=3 THEN MOVE 1000,50:DRA
W 900,150
880   IF hang=4 THEN MOVE 900,380:DRA
W 1060,380
890   IF hang=5 THEN MOVE 900,280:DRA
W 1030,380
900   IF hang=6 THEN MOVE 1060,380:DR
AW 1060,330
910   IF hang=7 THEN DRAW 1000,330:DR
AW 1000,290
920   IF hang=7 THEN DRAW 1040,290:DR
AW 1040,330:DRAW 1060,330
930   IF hang=8 MOVE 1060,290:DRAW 10
60,200
940   IF hang=9 THEN DRAW 1020,140
950   IF hang=10 THEN MOVE 1060,200:D
RAW 1100,140
960   IF hang=11 THEN MOVE 1060,270:D
RAW 1020,270:DRAW 1020,220
970   IF hang=11 THEN PRINT TAB(3,20)
"LAST"TAB(2,22)"CHANCE"
980   IF hang=12 THEN PROCdead
990   ENDPROC
1000
1010 DEF PROCdead
1020   MOVE 1060,270: DRAW 1100,270:DR
AW 1100,220
1030   VDU 19,0,11,0,0,0
1040   PRINT TAB(0,10)SPC(39)
1050   PRINT TAB(0,12)SPC(39)
1060   PRINT TAB(3,11)"The word is:~"
1070   PRINT TAB(6,14);word$(word)
1080   PRINT TAB(0,20) SPC(9):PRINT' S
PC(9)
1090   FOR J%=1 TO 20
1100     ENVELOPE 1,1, 0,0,0,0,0,12
6,-1,0,-3,126,126
1110     SOUND &10,1,3+RND(3),2
1120     FOR I%=0 TO 300+RND(2000):NEX
T I%
1130   NEXT J%
1140   TX=TIME:REPEAT UNTIL TIME-TX>10
0
1150   PRINT TAB(0,18)" Press"
1160   PRINT"" SPACE"

```

```

1170 PRINT " BAR "
1180 PRINT " for "
1190 PRINT " another "
1200 PRINT " word "
1210 REPEAT UNTIL GET=32
1220 game=1
1230 ENDPROC
1240
1250 DEF PROCright
1260 FOR P%=1 TO LEN(A$)
1270 IF MID$(A$,P%,1)=X$ THEN PRIN
T TAB(P%/3,7) X$:good=good+1
1280 NEXT P%
1290 U$=U$+X$
1300 guess=1
1310 SOUND 1,-15,101,2
1320 SOUND 1,-15,117,2
1330 SOUND 1,-15,129,2
1340 SOUND 1,-15,149,2
1350 ENDPROC
1360
1370 DEF PROCwrong
1380 PRINT TAB(2,10)"Wrong letters:-
"
1390 IF try<10 THEN PRINT TAB(try*2-
1,12) X$
1400 IF try>9 THEN PRINT TAB(try*2-1
9,13) X$
1410 try=try+1
1420 SOUND 1,-15,49,5
1430 SOUND 2,-15,45,5
1440 PROCgallows
1450 hang=hang+1
1460 ENDPROC
1740 REM ** list of easy words **
1750 DATA BOOK,SHOE,MORE,ROOM,SEA,CAME,GAME,MOVE,DATE,NICE,CAKE,CASE,TOOK,TAKE,
TIME,BELL,MODE,MAKE,COW,DOG,AND,FOR,GET,TREE,FIRE,FUN,BED,FROG,HOT,COLD,RUN,RAIN
,CUP,SHOP,TOY,TO,TOO,TWO,ONE,ROW,ROSE,EGG,SOCK,BOOT,BIKE,LOOK,BIRD,TOP,OUT,GO
1760 DATA END1
1770 REM ** List of hard words **
1780 DATA BECAUSE,VACUUM,AERIAL,TRIUMPH,NEGLECT,COMPUTER,BUCKET,DIFFERENCE,RADI
O,SOPHISTICATED,FAREWELL,WINDOW,CURTAIN,EXPLICIT,ECONOMY,WRAPPED,OXYGEN,DIMENSIO
N,INFINITE,ELEMENT,UMBRELLA,FENCE,ZEBRA,VISION,TURTLE
1790 DATA COLOUR,CARPET,OFFICER,CARTWHEEL,MALICIOUS,QUADRANT,HOOGLIGAN,FRECKLE,S
UGGEST,SPECIFIC,ROUTINE,MINIMUM,ASTERISK,PHIAL,SUBSTITUTE,POSITION,PHYSICAL,SUCC
UMB,VACATION,MIGRATE,DIALOGUE,DECIDUOUS,BUTTERFLY,PATIO,SKELETON
1800 DATA SANCTUARY,CUBICLE,GALAXY,ENCYCLOPAEDIA,CONDENSATION,PLUMBER,STATURE,C
ULINARY,AVIATION,BOTANICAL,HUMOROUS,DIPHTHERIA,MAGAZINE,DELIBERATE,BINOCULARS,BI
LLIARDS,ANOINT,ANTAGONIZE,PILGRIMAGE,PISTY
1810 DATA OFFICATE,OINTMENT,OMINOUS,PADDOCK,PALACE,PAGEANT,LIZARD,LOCU
ST,RAZOR,RECLINE,SACRILEGE,SANDWICH,SAUCE,THYME,TURNIP,TURBINE,UNCLE,ULCER,WALLO
W,WORSE,WRIGGLE,PLIGHT,POACH,GUARANTAGE,GUITAR,CORDIAL,ALTITUDE,ALLIGATOR,EXCITE
1820 DATA END2

```


RANDOM SELECTION

Based on a program and notes by JANE C. Tauranga

There are times when it can be useful to be able to select data items randomly without allowing repetition of any item. Such a facility as demonstrated in this program can be used to advantage in quiz type programs, to prevent the duplication of questions.

Each time you RUN the program, the list of data items will be printed out in a different order, without repetition of any item.

M% is the number of items in the DATA

lines 40-60 fills the array with these items

lines 80-110 select an item from the array and check to ensure it is not "Q" (eg it might be "cabbages")

line 120 prints the selected item ("cabbages"), then

line 130 changes "cabbages" to "Q" in the array. This line ensures that the original "cabbages" is not selected again - it no longer exists in the array, but has been replaced by an "Q".

10 MODE6	120 PRINTAB(10,3+2*I%)A\$
20 M%=5	130 word\$(pointer%)="Q"
30 DIM word\$(M%)	140 NEXT I%
40 FOR I%=1 TO M%	150 END
50 READ word\$(I%)	160 :
60 NEXT	170 DATA carrots
70 FOR I%=1 TO M%	180 DATA leeks
80 REPEAT	190 DATA cabbages
90 pointer% = RND(M%)	200 DATA beans
100 A\$ = word\$(pointer%)	210 DATA peas
110 UNTIL A\$ <> "Q"	

A MINOR REFINEMENT:

In the listing above, every time you add in more data, you have to remember to increase the value of M% in line 20 by the number of new items. You may, after a while, lose track of the number of items, so why not let the computer tot them up for you?

Lines 20-25 below could be added, plus line 999.

M% is set to -1 because after it has read each item, it will increase M% by one, before testing to see if the item just read is "end_of_list" or not. "end_of_list" is not an item of data you want included in your quiz, but it is a convenient marker for the computer

to know when to stop. If you don't have this last DATA statement (#999 here for my convenience!), when RUNNING the program it will crash, with the message 'Out of Data'.

```

20 M%=-1
21 REPEAT
22   READ A$
23   M%=M%+1
24   UNTIL A$="end_of_list"
25 RESTORE
999 DATA end_of_list

```

PROCSTAR

Gavin offers this procedure to demonstrate a PROC which draws star shapes.

X%, Y% are the coordinates of the centre of the star.

R1, R2 are the inner and outer radii (distance from the centre to the inner and outer points)

N% is the number of outer points, this is often set to 5 or 6.

Note the use of LOCAL which prevents corruption of variables used in the segment of code which calls the PROC, by variables of the same name used within the PROC. (eg the use of "I%" - try deleting line100 then RUN!)

```

10 MODE 4
20 FOR I%= 1 TO 9
30   R1=16*RND(32)
40   R2=R1*(1+3*RND(1))
50   PROC.star(RND(1280),RND(1024),R1,R2,4+RND(6))
60 NEXT
70 END
80
90 DEF PROC.star(X%,Y%,R1,R2,N%)
100 LOCAL A,D,I%
110 VDU29,X%,Y%;
120 MOVE R1,0
130 A = 0
140 D = PI/N%
150 FOR I% = 0 TO N%
160   A= A + D
170   DRAW R2*COS(A),R2*SIN(A)
180   A= A + D
190   DRAW R1*COS(A),R1*SIN(A)
200 NEXT I%
210 VDU 26
220 ENDPROC

```

Businessworld

COMPUTERS LIMITED

SPECIAL OFFER THIS MONTH

12 Month warranty on BEEBs bought in April

Businessworld is your only Wellington based source of service.

Ask us about:

Service Contracts
Cheap Monitors (some below cost)

Businessworld Computers Limited
11 Johnston Street.
WELLINGTON.

DATA GENERATION

21

Jeff H. Wgtn

Data retention poses problems on the Beeb which destroys its variables

- i) when BREAK is pressed
- ii) when the program is altered
- or iii) when RUN, CLEAR, or NEW are used.

When 'lost' the program can usually be resurrected with OLD but not the data. Even when the program stops without destroying data (eg at ESCAPE, STOP, or END or where no further line has yet been written), getting going again can be tricky.

The Beeb has no CONTINUE, and RUN destroys data, so GOTO must be used (surely the only case of the Beeb using GOTO while lesser machines don't?). You cannot GOTO a PROCedure even if you left from one (No proc at line...), nor a REPEAT UNTIL loop (No repeat at...). I may be wrong, but it appears that in a structured program, the only way back is GOTOing a line after DIMensioning but before the first REPEAT.

Since the line must actually exist, it is useful to make special note of the best place to re-enter a program without reinitialising, and also to avoid later crashing. This may not be a bad thing if it forces the programmer to think about the layout of his program, but far worse problems arise where:

- i) copious data is gathered throughout the program;
- ii) a large 'data rich' program is being developed and tested.

All of this may sound trivial to writers of small or low-data programs; but some programs may be using hundreds of names and numbers; all of which may need to be loaded to coax the computer along, until a missing comma necessitates reloading. Clearly the development and testing of such programs calls for data development techniques.

A simple approach:

```
100 REM INPUT A TO BE REINSTATED
105 A=6:REM to be deleted or
105 READ A:REM to be deleted (with appropriate error
data) or
105 A=RND(6):REM to be deleted
```

Beware of DATA though - I spent an hour debugging a perfectly good piece of code that was getting its data in a different order to the way I thought it was getting it.

An improvement could be:

```

20 TEST%=TRUE:REM to be deleted or
20 PRINT"Test mode? Y/N"
21 IF GET$="Y" THEN TEST = TRUE ELSE TEST = FALSE
followed elsewhere by:
510 IF NOT TEST THEN INPUT NAME$ ELSE NAME$="FRED"

```

Using TEST like this has advantages in that it could be left in for future testing and development, and could be modified to 'dummy run' a program. An obvious application is a game which displays until a player starts to play:

```

90 TEST=TRUE
100 PRINT"Press Y to play"
110 IF INKEY$(300)="Y" THEN TEST=FALSE

```

Perhaps the best solution is a device which produces fairly random data which will test the program to the full whilst not keeping it bogged down in its data validation routines. Since data can be numeric or string, I have produced two Functions which produce random data within a controllable range. The need for data generators was expressed in the January '83 newsletter. I hope this will go some way towards it.

The first produces a random number in the required range including negatives and with the required decimal point:

```

10 test%=TRUE
20 REPEAT
30 IF test% THEN INPUT,"Minimum, Maximum, Number of de
cimal places ",A,B,C%:REM to be made constants after te
sting by the tester!
40 IF test% THEN D=FNmakenumber(A,B,C%) ELSE INPUT D
50 PRINT D
90 UNTIL FALSE
100 END
110
120 DEF FNmakenumber(X,Y,Z%)
130 LOCAL lump%,answer
140 IF Y<=X THEN PRINT"Rubbish in Function":=1000
150 REPEAT
160 IF Y-X<1 THEN lump%=0 ELSE lump%=RND(INT(Y-X+1))-1
170 answer=X+lump%+(INT(RND(1)*10^Z%)/10^Z%)
180 UNTIL answer<=Y
190 = answer

```

The second gives a string in the required range and of the required length. Usually the range will be A to Z but 0 to 9 may be useful, and for example, in response to "Press Y to start" the values X,Z,I would do the trick without taking too long to find Y.

```

10 test%=TRUE
20 REPEAT
  60 IF test% THEN INPUT,"Low letter, High letter, Number
    of letters  "A$,B$,C%:REM to be made constants after test
    ing of the tester!
  70 IF test% THEN D$=Fmakestring(A$,B$,C%) ELSE INPUT D
$
  80 PRINT D$
  90 UNTIL FALSE
100 END
110
210 DEF Fmakestring(X$,Y$,Z%)
220 LOCAL answer$,N%,V$
230 FOR N%=1 TO Z%
240 V$=CHR$(ASC(X$)+RND(ABS(ASC(Y$)-ASC(X$))+1)-1)
250 answer$=answer$+V$
260 NEXT N%
270 =answer$

```

What I have offered here comes from the heart. I have a large, ugly, data-rich program resting on tape, waiting for a major load-up of data, and final debugging. I learned the above lessons too late, and there are no data generators in it. Ho hum, back to the keyboard.

INVERSE VIDEO Pt II

G.C.F.

This is the standard term for what last month I called Reverse Video. Here I refine some of the ideas therein, and add a couple more. By the way, did any of you notice the reference to MODE 8? Yes, it does exist, but you have to set it up yourself without using the Mode command. Details are in the Advanced User Guide.

BETTER INVERSE CAPITALS

The simple techniques for inverse video last month made it difficult to see the tops or certain capital letters like "E", as the top line of the letter merged into the normal background. The way round this is to add an extra line of 'inverse background'. This can only be done in graphics modes (ie 0,1,2,4 & 5) and then may "overwrite" part

of the line above. This latter problem can be avoided by only printing on alternate lines. (Or not using letters that have descenders!)

In order to draw this line we have to know:

- i) the graphics coordinates of first characters,
- ii) the number of characters in the string,
- iii) the width of a pixel in the current mode.

```

10 FOR I%=0 TO 2
20  MODE I%
30  PRINT"pqpy - letters with des
cenders - pqpy"
40  PROC_tidy_inverse("CAPITAL an
d lower case")
50  PRINT" SAME LINE"
60  PRINT"Yet another line"
70  PRINT
80  PRINT"Now press "
85  PRINT
90  PROC_tidy_inverse("RETURN")
100 PRINT" for the ";
110 PROC_tidy_inverse("next MODE"
)
120 dummy%=INKEY(3000)
130 NEXT I%
140 END
150
160 DEF PROC_tidy_inverse(A$)
170 LOCAL chr_width%,length%,mode%,
XL%,X_pixel%,XR%,Y%
180 mode% = (FNfx(&87,0,0) AND &FF0
000) DIV &10000
190 X_pixel% = 2 * 2^(mode% MOD 3)
:REM modes 0,1,2,4,5
200 chr_width% = 8 * X_pixel%
210 XL% = POS * chr_width%
220 length% = LEN(A$) * chr_width%
- X_pixel%
230 XR% = XL% + length%
240 Y% = (32 - VPOS) * 32
250 MOVE XL%,Y%
260 DRAW XR%,Y%
270 VDU17,0,17,135
280 PRINT A$;
290 VDU17,7,17,128
300 ENDPROC
310
320 DEF FNfx(A$,X%,Y%)=USR &FFF4

```

NB for an explanation of lines 180 and 320 see pp42-44 of Dec/Jan issue

INVERSE VIDEO - MODE 7

It does not seem possible to have black text on a white background. The closest would be red or blue text on white background. (Chapt. 28 of UG, pp152 especially).

INVERSE USER-DEFINED CHARACTERS

These can be included in strings and inverted in the same way as in Pt.I. An alternative is to redefine your normal character as its inverse. The following program first defines a diamond shape, then its inverse in two different ways, before displaying them.

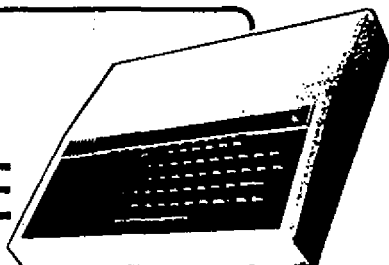
```

10 MODE4
20 VDU23,225,&08,&1C,&3E,&7F,&3E,&
1C,&08,&00
30 VDU23,235,NOT&08,NOT&1C,NOT&3E,
NOT&7F,NOT&3E,NOT&1C,NOT&08,NOT&00
40 VDU23,245,&7F,&3E,&C1,&00,&C1,&
E3,&F7,&FF
50 PRINT CHR$(225)
60 PRINT
70 PRINT CHR$(235)
80 PRINT
85 PRINT
90 PRINT CHR$(245)
100 END

```



BBC SOFTWARE



The MOLYMERX COLOUR SOFTWARE CATALOGUE is ready !

Pages of descriptions of new powerful Application and
Utility software for \$3.00

and THAT's NOT ALL -

as we publish new software we send you free UPDATES.

Our offer is to keep you abreast of NEW products,
NEW services and our regular special prices.

Some of the products in the latest edition of our catalogue !

DISK FILE

True Random Access File DATABASE

WAR GAMES

Napoleon Crusaders Emperor

SHUTTLE II

DISK ZAP

MONITOR

Fly COLUMBIA II from space to a Desert landing (if you can !)

Please use BLOCK CAPITALS

Name/Company

Address

..... Zone

I enclose \$..... @ \$4.50 per copy
TR8-80 Catalogue
I enclose \$..... @ \$3.00 per copy
Colour Catalogue

Australian distributors for: Aardvark, Acorn, Big Five, C.A.U.,
A.J. Harding, Hexagon, ICR Futuresoft, Logical Systems,
Micro Systems Software, Munford Micro Systems,
Pioneer Software, Understanding, ABC, Powersoft.

Dealer Enquiries Welcome

Fill in the coupon and return to:



Molymex Ltd.
P.O. Box 60-152
Thirangi,
AUCKLAND
NEW ZEALAND

(817) 4372

Please send ☐ copies of the
Molymex Software Catalogue.

INSTANT ARCADE GAMES

Reviewed by ANTON E. Wellington
 INSTANT ARCADE GAMES for the BBC Micro.
 Jean Frost, Pan Books (Collins), 1983. \$14.85 (Whitcoulls)

This book is one of a series giving Instant Arcade Games for the ZX Spectrum and Dragon 32. The book doesn't suffer too badly from its history. The fundamental attitude is to develop a set of building blocks, each covering a function: PROCrules, PROCshoot, etc. The back page blurb indulges in the buzzwords "structured programming", which is quite an exaggeration, but the essence is there. Break up the problem into units, write adaptable, clear and coherent code for each unit, then link the units together. The main control program shows the intention clearly, but the use of GOTO to create a loop, rather than a 'REPEAT.. ..UNTIL bored', is an unfortunate lapse.

Not a superb book, but adequate. It would make a good gift for a teenager. By the time he (or she) found the book boring, she (or he) should be writing sprites routines in machine code.

LOONEY LIFT

Reviewed by Richard J Richmond

LOONEY LIFT from H&H Software, 53 Holloway, Runcorn, Cheshire WA7 4TJ
 \$7.50+\$1.50 p&p airmail (80 track disc \$2.50 extra).

"Three more complaints and you're out," says the manager - part of the clear instructions that lead into the game. On 'space' you are presented with an opened up ten storey hotel like those buildings a stranger notices in Wellington.

Before you ("liftie") have time to look around, unfriendly guests depart their rooms and head 'hell bent' to avoid the lift and get a complaint into the manager.... Unless you can intercept them and deposit them, one at a time, on the ground floor.

There is however another complication - a conveyor belt along the ground floor bears suitcases to your lift. If you aren't there when the case arrives it simply drops down the shaft, to your cost.

For a first arcade type game we found it was well presented, had good sound effects, was colourful (a colour screen helps but is not essential), fast as you like and enjoyable - challenging even.

The User Group has H&H's software catalogue. It runs to several pages so if you want more detail on their offerings plus prices, send us \$1 to cover photocopying plus mailing.

H&H's range is mostly educational, suitable for both home and school use. There are programs for children just starting to read, designing tessellatable shapes, running a chemicals plant.... Airmail p&p from H&H is \$1.50 over the full price.

BEEBUG SUBSCRIPTIONS

Seventy of our members have joined our bulk subscription to Beebug and many have been disappointed by the erratic supply situation. We apologise to all of you and especially to those who have not received anything yet. The service from the UK has been most unsatisfactory, with poorly packaged consignments requiring repackaging by the Post Office, and arriving surface rather than air mail. Consignments have generally been incomplete and we have not received any November editions. We are writing to Beebug, acknowledging receipt of everything that reaches us and reminding them of what is still missing. I hope we can get this problem cleared up soon!



BENNETTS

AND THE BBC

Biggest Range of books on the BBC microcomputer in N.Z.

Wide range of cassettes and disc software from ACORNSOFT and other leading software houses

Exciting range of BBC microcomputers and accessories
(Joysticks, printers, video monitors etc.)

Huge range of general computer books

Ask for our Computer Booklist and BBC Accessory/Software Catalogue

BENNETTS BOOKSHOP phone 83809
BOX 138 telex N23649
PALMERSTON NORTH

ADVANCED PROGRAMMING

Reviewed by TIM R.

Jim McGregor & Alan Watt. Advanced Programming Techniques For The BBC Micro pub. Addison Wesley. 365p. \$29.95
Copy supplied for review by Whitcoulls.

This is definitely a book for those proficient in BASIC programming - it expects quite a high knowledge of BBC BASIC, and some mathematical knowledge for one of the chapters. However it is absolutely packed with great programming tips and ideas on a wide variety of subjects. All the programs are written using a very high standard of programming and very good structure. Variable names are almost always meaningful, especially the PROC labels. The book is split into ten chapters, five appendices, and quite a good index. Each chapter is separate from its neighbours which means that the book does not have to be read from the start so that any particular chapter can be understood. I hear the book was developed from a series of lectures which would explain the 'modularity' of the sections.

It would be ridiculous to try to list all the topics covered in the book as there are so many of them. Here are a few to whet your appetite.

LOGICAL PROCESSING OF COLOUR AND INTERACTIVE GRAPHICS
THREE DIMENSIONAL GRAPHICS
ADVANCED USE OF SOUND
BOARD GAMES AND GAME TREES

... and so it goes on.

The appendices are helpful lookup tables for some of the topics discussed in the main section of the book, and further discussions of the techniques used. It would be hard to praise this book highly enough - it is packed with a large supply of 'goodies' that can be directly applied to user programs. I have tried a large percentage of the programs in the book and they all ran immediately (once I removed my typos) and worked as expected.

I have only two relatively minor complaints - both on the book layout. The authors follow in the footsteps of all those people who have never heard of the LISTO listing formatter function. Doesn't anybody but I and the NZ User mag use LISTO ?? It makes formatting the listing totally automatic, and is far simpler than doing the indenting oneself which is obviously what has been done in this book! The other complaint I have is the typestyle. It is not as bad as dot matrix, but it is certainly not proportionally spaced, and appears to have been done on a daisywheel printer. Readable but not remarkable !

WHO IS IT FOR?

I would recommend it as very useful for those with some skill in writing BASIC who want to add some flair to their software - especially in the area of game design, though certainly not only that. (And it's got more on Sound, beyond their first book's coverage! -Ed)

BEYOND BASIC

Reviewed by Anton E. Wellington

BEYOND BASIC 6502 Assembly Language Programming for the BBC Microcomputer
 Richard Freeman
 BBC Publications, London, 1983. \$25.50 (Whitcoulls)

Finally a book I like! I've liked others, but never got to review them. I own Birnbaum's book on assembly language, and members may be interested to know that Birnbaum gives a more extensive coverage, and details more complex examples than Freeman. Yet I would first recommend Freeman to anyone who wishes to get a clear introduction to assembly language. I've bought a copy myself.

The layout is very explicit, with subheadings boldly laid out, and important definitions are emphasised by boxes and green print. The book lacks an index, but this appears to be a production problem, as an index is referred to on page 8. The only major problem I noticed was at the top of page 95, for the list of instructions using absolute indexed mode. Indexed Y cannot be used interchangeably with indexed X, and the table provided on page 252 should be used in preference to the erroneous list.

The high point of the book is a sort routine that takes up to 256 strings in a basic array, and sorts them in about 5 seconds. The program makes use of a parameter block set up by BASIC when CALL is executed. By passing the array in as a parameter, the location of the array placed in page &600. So, no need to find the array, or specially put it somewhere. Simply make good use of the features supplied.

All the programs are on tape, which can be bought as an optional extra. The stringsort program is 14 blocks long, (about 5000 characters),

and members can type from the book, or buy a copy. It depends whether you like learning from typing and debugging, or from use. I do recommend this book.

SPECIAL GENERAL MEETING

THURSDAY 29 MARCH 1984 7.30 pm

at the Correspondence School Staffroom, Portland Cres, Thorndon, Wellington.

You will all have read the statement sent to you by our President, Warren Willis, advising members of the Special General Meeting scheduled for latter this month. To recap -

THE MOTIONS:

1. That clause 1g of the Constitution of the BBC Microcomputer Users Group of NZ Inc be altered to read:
"The financial year of the group shall be in any year from 1 January to 31 December of the same year."
2. That the financial year shall run from 1 April 1984 to 31 December 1984.
3. That the 1984 subscription for the period 1 April 1984 to 31 March 1985 be set at \$30.00 annual fee plus \$5.00 joining fee, subject to ratification at the forthcoming 1984 AGM (by which time the 1983/84 accounts will be audited and available).

COMMENTS:

The first two motions are technical adjustments to facilitate the financial administration of the society. The third motion can be considered in two parts -

1. It offers an interim solution to the current hiatus between the end of the subscription year and the setting and receipt of next year's subscription.
2. It proposes that the subscription be increased to \$30 .

The Constitution provides the Committee with discretion to waive part or all of this rate where this is considered appropriate. Members may be interested to note that another computer society with which I am associated provides in addition a specific concessional rate for students, unemployed people and others with limited means (eg pensioners).

Finally, I would suggest that members unable to attend the S.G.M. may wish to send in their views to the President. The Constitution also states, in clause 7e, that "any financial member unable to attend a meeting may cast a vote in writing on a previously tabled or circularised motion." Please remember to include your membership number in any correspondence.

HALL OF FAME			
PUBLISHER:	GAME	HIScore	SCORER
Acorn	Arcadians	15,000	Felix G.
	Countdown	127	Tim R.
	Hopper	20,900	Brett A.
	Meteors	18,000	Felix G.
	Monsters	50,320	Brian G.
	Philosophers Quest	236	Judy
	Planetoid	378,950	Stevie H.
	Rocket Raid	140,777	Mark R.
	Snapper	250,000	Susan
	Starship Command	642	Steve W.
	Bug Blaster	37,642	Steve W.
Alligat.	'747'	205	Steve W.
Dr. Soft	Gorf	18648	Steve W.
FG Soft.	Mutant Defender	6,900	Steve W.
Level 9	Colossal Cave	687	Steve W.
	Dungeon Adv.	330	Anne
	Snowball	550	Gavin
Prog.Pow	Danger(1x8,Lvl 1)	82789	Steve W.
	Killer Gorilla	91,000	Susan
	Super Invaders	19610	Brett A.

Can we help you with items for your B.B.C. ?

SOFTWARE now in stock:-

Most of the BBCSOFT and ACORNSOFT ranges

The full I.J.K. range

MOLYMERX JUMBO and SPACE SHUTTLE

For serious applications consider the GEMINI range-

BEEBCALC BEEBLOT DATABASE MAILING LIST

STOCK CONTROL HOME ACCOUNTS EASILEDGER

also CASHBOOK and FINAL ACCOUNTS

Just arrived is the impressive AB2 Designs Package

HARDWARE Now Available:-

Disc Drives 80 track Double Sided: \$800

(these can be supplied with 40 track switch)

Printers: \$750

Acorn Joysticks (pair): \$55

Programmable Keypad, 16 Key: Available soon.

For further information contact:

Bob Brady at

phone 439016

Pirimai Pharmacy Ltd.

Pirimai Plaza

NAPIER

USER GROUP MEETINGS

AUCKLAND:-meets 2nd Wednesday of the month, 8.00pm, VHF clubrooms, Hazel Ave, Mt Roskill. ph Dave 770-630 x 518(bus) or Kerry 695-355.

HAMILTON:-last Thursday of the month 4.45pm Waikato Tech B-block staffroom. Ph. Peter, Hamilton 392-500 x 897(bus) or Alison, Morrinsville 6695.

TAURANGA:-every second Wednesday. ph. Chris & Jane 65-076.

HAWKES BAY:-Alternative months Hastings & Napier. Contact Kendall Napier 435-624, Bob Taradale 446-955, Mitch Hastings 778-235.

TAUPO:-CONTACT Dave 84-215.

WELLINGTON:-meets LAST Thursday of the month, 7.30pm, Correspondence School staffroom (1st floor), Portland Cres, Thorndon. ph. Anton 286-289 or Warren 787-005.

CHRISTCHURCH:-fortnightly workshops, Monday 6.30pm at Hagley High School. ph. Michael 582-267.

OTHER CENTRES:- let us know and we will publish details here.

THE NEWSLETTER:

CONTRIBUTIONS:-most welcome. Listings should be sent in on tape or disc.(Tape - 2 copies, one at 300 baud please). A 35c stamp gets your tape / disc returned too. Please include written explanation of listing; text only material welcome too.

DEADLINES:- For a particular issue, the last day for material is the last day of the month prior to publication.

ADVERTISING:- Rates are \$20 per half page, deadline as above.

Copyright (c) THE BBC MICROCOMPUTER USERS GROUP OF NZ INC.
Published monthly except January and mailed to financial members.

MEMBERSHIP:

Membership of the users group is on payment of an annual subscription from April to March. For the 1983/4 year it is: \$20.00, plus a \$5.00 joining fee for new members, plus \$5.00 for Auckland & Wellington members as a local branch levy, which entitles attendance at meetings as detailed above. Renewing members get all back issues of the current year; New members get ALL back issues to v.1 #1 Jan '83.

THE BBC MICROCOMPUTER USERS GROUP OF NZ INC
P.O.Box 9592
Wellington

Reproduced by ROBERTS PRINTING Co. Ltd, Wgtn.